Sugeno Integral for Rule-Based Ordinal Classification

Quentin Brabant, Miguel Couceiro, Didier Dubois, Henri Prade, Agnès Rico

Sugeno Integral for Rule-Based Ordinal Classification

Quentin Brabant[1], **Miguel Couceiro**[1], **Didier Dubois**[2], **Henri Prade**[2], **Agnès Rico**[3]
(1) Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
(2) IRIT, CNRS, Université Paul Sabatier 118 route de Narbonne 31062 Toulouse
(3) ERIC, Université Claude Bernard Lyon 1, 43 bld du 11-11, 69100 Villeurbanne
{quentin.brabant, miguel.couceiro}@loria.fr, {dubois, prade}@irit.fr, agnes.rico@univ-lyon1.fr

Abstract

We present a method for modeling empirical data by a rule set in ordinal classification problems. This method is nonparametric and uses an intermediary model based on Sugeno integral. The accuracy of rule sets thus obtained is competitive with other rule-based classifiers. Special attention is given to the length of rules, i.e., number of conditions.

First author note :

Because of a mistake I made, the results presented in the original paper are partly wrong. This document is a corrected version.

1 Introduction

Let $\mathbf{X} = X_1 \times \cdots \times X_n$, where each X_i is a totally ordered set called *attribute* domain, and let L be a totally ordered set, whose elements are called *classes*. The minimal and maximal element of any totally ordered set X are denoted by 0_X and 1_X, respectively. An *instance* is a pair $(\mathbf{x}, y) \in \mathbf{X} \times L$. A *dataset* is a collection of instances (in which the same instance can appear several times). A *model* of a dataset is a function $f : \mathbf{X} \to L$, and its accuracy is the proportion of instances for which $f(\mathbf{x}) = y$ in agreement with the dataset.

We consider datasets that can be accurately modeled by a nondecreasing function. Such datasets are typically found in Multi-Criteria Decision Aid, where evaluation of alternatives depends on several criteria, but also in some medical diagnosis problems. The task of finding an accurate nondecreasing model of a dataset has been addressed in several ways (see, e.g., [Gutiérrez *et al.*, 2016]). In this short paper, we focus on rule-based models, since rules provide an explicit justification for each class prediction they make. We consider sets of (selection) rules of the form

$$\textbf{if } x_1 \geq \alpha_1 \textbf{ and } \ldots \textbf{ and } x_n \geq \alpha_n \textbf{ then } y \geq \delta \qquad (1)$$

where $(\alpha_1, \ldots, \alpha_n) \in \mathbf{X}$. The VC-DomLEM algorithm [Blaszczyński *et al.*, 2011] allows us to learn such a set of rules, which yields a good accuracy compared to other interpretable models.

In [Brabant *et al.*, 2018], we proposed an alternative method for learning rule sets, which relies on Sugeno integrals. This method does not require the tuning of any hyperparameter and is competitive with VC-DomLEM in terms of accuracy. Moreover, this method raised new questions about the relevance of capacities (i.e., monotonically increasing set functions) in data-modeling.

2 Rule-based and capacity based models

Let R be a set of rules of the form (1). There may be several functions that are compatible with R. We denote by f_R the smallest function compatible with R, defined by $f_R(\mathbf{x}) = \max_{r \in R} f_r$ such that for each rule r:

$$f_r(\mathbf{x}) = \delta^r, \textbf{ if } \forall i \in C,\ x_i \geq \alpha_i^r, \textbf{ and } 0 \textbf{ otherwise.}$$

We will say that a function f is *equivalent* to R if $f = f_R$.

In what follows, we use the notation $[n] = \{1, \ldots, n\}$. Let $\mu : 2^{[n]} \to L$ be a *capacity*, i.e., a set function $2^{[n]} \to L$ such that $\mu(\emptyset) = 0_L$, $\mu([n]) = 1_L$, and $\mu(I) \leq \mu(J)$ for all $I \subseteq J \subseteq [n]$. The Sugeno integral w.r.t. μ is the aggregation function $S_\mu : L^n \to L$ defined by

$$S_\mu(x_1, \ldots, x_n) = \max_{I \subseteq [n]} (\min(\mu(I), \min_{i \in I} x_i)).$$

Note that the Sugeno integral can be a model for ordinal classification only if $X_1 = \cdots = X_n = L$. A *Sugeno utility functional (SUF)* is a more expressive model which can merge values from different scales. Let $\varphi = (\varphi_1, \ldots, \varphi_n)$, where each $\varphi_i : X_i \to L$ is a nondecreasing function such that $\varphi_i(0_{X_i}) = 0_L$ and $\varphi_i(1_{X_i}) = 1_L$. The SUF $S_{\mu,\varphi}$ is the function defined by

$$S_{\mu,\varphi}(x_1, \ldots, x_n) = S_\mu(\varphi_1(x_1), \ldots, \varphi_n(x_n)).$$

It was shown in [Brabant *et al.*, 2018] that:

1. Any SUF is equivalent to a rule set.
2. Any single rule is equivalent to a SUF.
3. Some rule sets are not equivalent to a single SUF.

In other words, some combinations of rules cannot be expressed by one SUF. However, the second assertion allows to say that any rule set is equivalent to some function $M_{\boldsymbol{S}} : \mathbf{X} \to L$ defined by $M_{\boldsymbol{S}}(\mathbf{x}) = \max\{S_{\mu,\varphi}(\mathbf{x}) \mid S_{\mu,\varphi} \in \boldsymbol{S}\}$, where $\boldsymbol{S}$ is a set of SUFs. We call such function a max-SUF.

There is no reason to think that a max-SUF provides a better interpretability than its equivalent rule set. However, an interesting question is whether it can serve as an intermediary model that helps guiding the learning process of a rule based model. Indeed, in [Brabant *et al.*, 2018], it is shown that a non-parametric learning algorithm based on a max-SUF is competitive with VC-DomLEM.

3 From SUFs to rule sets and vice-versa

Any SUF $S_{\varphi,\mu}$ is equivalent to the rule set

$$\bigcup_{I\subseteq[n]} \bigcup_{\delta\leq\mu(I)} \{\forall i \in I,\ x_i \geq \alpha_i \Rightarrow y \geq \delta\}, \tag{2}$$

where $\alpha_i = \min\{a \in X_i \mid \varphi_i(a) \geq \delta\}$. Note that this set is likely to contain redundant rules. Now let us show a method of translation of a rule set R into a SUF.

1. Initialize μ and $\varphi = (\varphi_1, \ldots, \varphi_n)$ with minimal values.
2. For each rule $x_1 \geq \alpha_1, \ldots, x_n \geq \alpha_n \Rightarrow y \geq \delta$ in R:
   (a) let $A = \{i \in [n] \mid \alpha_i > 0\}$,
   (b) increase $\mu(A)$ up to δ,
   (c) for each $i \in A$, increase $\varphi_i(\alpha_i)$ up to δ

After these steps we always have $S_{\mu,\varphi} \geq f_R$. When $S_{\mu,\varphi} > f_R$, no SUF is equivalent to R.

In some cases, it is not problematic that $S_{\mu,\varphi} > f_R$. For example, if f_R is a model of a dataset $\mathcal{D}$, we may want to find an SUF that best fits $\mathcal{D}$. Obtaining $S_{\mu,\varphi} = f_R$ is not always possible since SUFs are not expressive enough. However, equality can be always achieved using a max-SUF [Brabant *et al.*, 2018]. The method presented in the next section relies on this fact.

4 Learning rules from empirical data

Let $\mathcal{D}$ be a dataset. The following three steps provide a method for modeling $\mathcal{D}$ by a max-SUF.

1. Selection of an order-preserving subset of data. Two instances $(\mathbf{x}, y)$ and $(\mathbf{x'}, y')$ can be *anti-monotonic* together, i.e, $\mathbf{x} \leq \mathbf{x'}$ and $y' \leq y$. We iteratively remove instances from $\mathcal{D}$, starting from those that are anti-monotonic with the highest number of other instances, until no anti-monotonic pair remains. We denote by $\mathcal{D}^-$ the dataset obtained in this way.

2. Modeling $\mathcal{D}^-$ by a rule set R. Initialize R to $\emptyset$. For each instance $((a_1, \ldots, a_n), y)$ in $\mathcal{D}^-$, search for $A \subseteq [n]$ with minimal cardinality, such that the rule

$$\forall i \in A,\ x_i \geq a_i \Rightarrow y \geq \delta, \tag{3}$$

is not contradicted by any instance in $\mathcal{D}^-$. Add the rule (3) to R. At the end of this step, the class of each instance in $\mathcal{D}^-$ is exactly predicted by f_R.

3. Translation of R into a max-SUF. See Algorithm 1. The obtained max-SUF is not necessarily equivalent to R, but it fits $\mathcal{D}^-$ precisely.

Note that the max-SUF given by this method can be translated back into a rule set, which constitutes an equivalent model and is easier to interpret.

Algorithm 1: Makes a partition $\mathbf{P}$ of R such that the max-SUF M_S verifies $M_S(\mathbf{x}) = y$ for each instance $(\mathbf{x}, y)$.

```
P ← {}
for each r ∈ R do
    affected ← false
    for each P ∈ P do
        translate P into a SUF S_{μ,φ}
        if S_{μ,φ}(x) ≤ y for all instance (x, y) in D⁻ then
            add r to P
            affected ← true
            break loop
    if affected = false then
        add {r} to P
```

	1	2	3	4	5	6	7	8	9	10	11	12	avg.
Steps 1,2.	74.4	95.8	97.7	93.6	91.7	65.6	83.2	27	67	63.6	58.2	51.4	72.4
Steps 1,2,3.	76	95.3	97.2	89.3	92.4	65.2	84.5	26.4	69.4	63	56.7	53.2	72.4
VC-DomLEM	76.7	96.3	97.1	91.7	95.4	67.5	87.7	26.9	66.7	55.6	56.4	54.6	72.7

Table 1: Accuracy obtained with each method on each dataset. Datasets are numbered as in [Blaszczyński *et al.*, 2011]

5 Empirical study

We compared our method to VC-DomLEM on the 12 datasets in [Blaszczyński *et al.*, 2011]. In order to show the importance of Step 3 in our method, we separately evaluated the rule set given by steps 1 and 2 alone, and the max-SUF given by steps 1,2, and 3. We see that Step 3 do not increases the accuracy on average. Therefore, the good results of this method are not due to the use of SUFs, but to the 2 first steps.

The *length* of a rule is the number of attributes X_i where $\alpha_i > 0_{X_i}$ (since the condition $\alpha_i \geq 0_{X_i}$ is trivial). Shorter rules are easier to interpret and constitute more concise models. Table 2 shows the rule length distribution obtained after steps 1, 2, and 3. The dual of max-SUFs are the min-SUFs that correspond to sets of (rejection) rules of the form

$$\textbf{if } x_1 \leq \alpha_1 \textbf{ and } \ldots \textbf{ and } x_n \leq \alpha_n \textbf{ then } y \leq \delta.$$

When learning min-SUFs by a dual method, the rule-length distribution differs from that obtained by learning max-SUFs. Long rules of one type sometimes go along with short rules of the other type.

References

[Blaszczyński *et al.*, 2011] J. Blaszczyński, R. Slowiński, and M. Szelag. Sequential covering rule induction algo-

Rule length

dataset	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		1	64	23	11	1										
2	13	72	13	1	1	1	1	1								
3			3	24	41	32										
4	38	47	10	3	1	1										
5	3	25	31	13	5	2	3	2	1	1	3	5	3	1		1
6	2	20	37	22	6	2	1	1	1	9						
7		20	21	21	12	9	15	3								
8	6	62	16	16												
9	6	26	36	32												
10	5	26	42	27												
11	6	2	25	45	9	5	4	1		3						
12	3	24	44	21	5	1				2						

Table 2: Percentage of rules with a given length, for each data set.

rithm for variable consistency rough set approaches. *Information Sciences*, 181(5):987 – 1002, 2011.

[Brabant *et al.*, 2018] Q. Brabant, M. Couceiro, D. Dubois, H. Prade, and A. Rico. Extracting decision rules from qualitative data via Sugeno utility functionals. In *Proc. Int. Conf. on Inf. Proc. & Manag. of Uncertainty, Cadiz, Spain (IPMU'18)*, 2018.

[Gutiérrez *et al.*, 2016] P. A. Gutiérrez, M. Pérez-Ortiz, J. Sánchez-Monedero, F. Fernández-Navarro, and C. Hervás-Martínez. Ordinal regression methods: Survey and experimental study. *IEEE Trans. on Knowledge and Data Engineering*, 28(1):127–146, 2016.